REMARKS

Claims 1-20 are pending in the present application. Claim 1, 10, and 17 have been amended to provide a more consistent reading by changing "the store" to "the mispredicted path side memory" and by referring to the "in parallel" language as in parallel to the instruction pipeline stages. Reconsideration of the rejection of the application is respectfully requested in view of the following remarks.

The Claims Are Allowable Over the Prior Art

Claims 1-14 and 17-20 were rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 5,860,017 to Sharangpani ("Sharangpani").

As discussed previously, embodiments of the present invention refer to branch misprediction recovery. As recited in each of the pending claims, an instruction that is predicted not to be executed is advanced through an instruction pipeline. Results of the processing in the instruction pipeline can be stored in a mispredicted path side memory in parallel to the instruction pipeline. In the case of a branch misprediction, the results that are stored in the mispredicted path side memory can be restored to the instruction pipeline.

In the current Office Action, the rejection of the claims is similar to the previous Office Action. There is, however, a "Response to Arguments" section beginning on page 5. The section states that it is well known to provide buffers, registers and memory to store results of instruction processing. Certainly, results of instruction processing in an instruction pipeline

stage must be maintained in some manner so that they are provided to the next stage. The pending claims, however, refer not only to storing, but storing results in parallel to the instruction pipeline stages and restoring results from the mispredicted path side memory to the instruction pipeline.

An example of this is shown in Table II and the description at pages 5-8. As seen in Table II, instructions from a branch predicted not to be taken (XXX2 to XXX4) are processed in the instruction pipeline stages. Instructions (YYY1-YYY6) from the branch predicted to be taken are also processed in the instruction pipeline stages. At time "6", instruction YYY1 is processed by stage one, and the results of stages two through four for instructions XXX2 to XXX4 are stored. At time 7, instruction YYY1 is processed through stage two and instruction YYY2 is processed through stage one. That process continues at times 8-11 where stages one through four are processing instructions YYY1-YYY6. Once the branch instruction JCC-Y1 is executed, it is learned that the branch is mispredicted and the processing of instructions YYY1-YYY6 is no longer needed. At time 12, the results of stages two through for instructions XXX2 to XXX4 can be restored to the pipeline stages for further processing and execution.

Sharangpani refers to speculative execution of instructions. As indicated at Col. 3, line 41 to Col. 4, line 30, the processor looks for front-end and back resources that are available and inserts instructions from a second target into the execution pipeline. As indicated at Col. 4, lines 23-30, "[o]nce the condition of the conditional branch instruction is resolved, instructions from the incorrect stream are canceled while instructions from the correct instruction stream continue through any additional processing stages. Valid, executed instructions are subsequently retired and committed to architectural state. In this manner, the performance penalty incurred for

branch mispredictions is significantly reduced if not eliminated." Though intermediate results of stages in Sharangpani are stored in registers, buffers, memory, etc., those memory devices are part of the pipeline stages. There is nothing in Sharangpani that suggests that the results of pipeline stages are to be treated differently depending on whether the instructions processed by those stages are from predicted-taken branches or predicted-not-taken branches. When a branch has been mispredicted, results of the predicted branch are canceled, and the results from the mispredicted path continue execution.

The Office Action concedes that Sharangpani does not expressly detail "storing in a mispredicted path side memory in parallel a result of the instruction in the instruction sequence that is predicted not to be executed from the instruction pipeline, and restoring in parallel from the store in to the instruction pipeline for continued execution if an instruction sequence predicted to be executed is mispredicted." (Office Action p. 2). Since instructions from the mispredicted branch are treated in the same manner as instructions from predicted branches, Sharangpani does not suggest such a feature either. The storing and restoring operations shown in Table II are not suggested anywhere in Sharangpani. In Sharangpani, instructions XXX2-XXX4 would continue through processing stages during times 7 to 12 (i.e., there would be no storing in parallel of the results to a mispredicted branch side memory). Once the branch is determined, these instructions would continue to be processed (i.e., there would be no restoring of results from the mispredicted branch side memory).

Though the "use of buffers to buffer results at each stage of processing are well known to be used in conventional processing pipelines" (Office Action, p. 6), there is nothing in Sharangpani that teaches or suggests that such buffers are to be used to store results in parallel to the instruction pipeline or that they be used to restore results from these buffer into the

instruction pipeline. Sharangpani only teaches that such buffers would be used in the same manner for processing instructions from the predicted and mispredicted branches.

In view of the above, reconsideration and withdrawal of the rejection of claims 1-14 and 17-20 under 35 U.S.C. § 103(a) is respectfully requested.

CONCLUSION

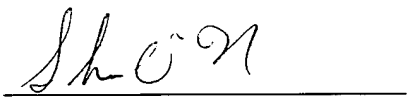
Applicant respectfully requests entry of the above amendments and favorable action in connection with this application.

The Examiner is invited to contact the undersigned to discuss any matter concerning this application.

The Office is hereby authorized to charge any fees required under 37 C.F.R. §§ 1.16 or 1.17 or credit any overpayment to Kenyon and Kenyon Deposit Account No. 11-0600.

Respectfully submitted,

KENYON & KENYON LLP



Shawn W. O'Dowd
Registration No. 34,687
Attorney for Intel Corporation

Date: June 23, 2006

KENYON & KENYON LLP
1500 K Street, N.W.
Suite 700
Washington, D.C. 20005
Ph.: (202) 220-4200
Fax.: (202) 220-4201
DC1-617715